UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/758,501	01/16/2004	Donald Edward Benson	SVL920030128US1	6340

46159 7590 08/04/2010

SUGHRUE MION PLLC
USPTO CUSTOMER NO WITH IBM/SVL
2100 PENNSYLVANIA AVENUE, N.W.
WASHINGTON, DC 20037

EXAMINER	
STACY, BRENT S	

ART UNIT	PAPER NUMBER
2161	

MAIL DATE	DELIVERY MODE
08/04/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte DONALD EDWARD BENSON, EDWARD JOSEPH GALLAGHER, MANG-RONG HO, and DWAYNE LORENZO RICHARDSON

Appeal 2009-002610
Application 10/758,501
Technology Center 2100

Before JOHN A. JEFFERY, JAY P. LUCAS, and JAMES R. HUGHES, *Administrative Patent Judges*.

JEFFERY, *Administrative Patent Judge*.

DECISION ON APPEAL[1]

Appellants appeal under 35 U.S.C. § 134(a) from the Examiner's rejection of claims 1-4 and 7-20. We have jurisdiction under 35 U.S.C. § 6(b). We affirm.

---

[1] The two-month time period for filing an appeal or commencing a civil action, as recited in 37 C.F.R. § 1.304, or for filing a request for rehearing, as recited in 37 C.F.R. § 41.52, begins to run from the "MAIL DATE" (paper delivery mode) or the "NOTIFICATION DATE" (electronic delivery mode) shown on the PTOL-90A cover letter attached to this decision.

STATEMENT OF THE CASE

Appellants' invention efficiently validates and processes requests based on the respective versions of affected items and associated objects. *See generally* Abstract. Claim 1 is illustrative:

> 1. A method of validating a request in connection with an interactive content database, said method comprising:
>
> receiving a request that affects an item;
>
> identifying a version of the item based on a first time;
>
> determining whether the request affects an object associated with the item;
>
> identifying a version of the object based on a second time when the request affects the object, wherein identifying the version of the object comprises retrieving a timestamp and an identifier for the object; and
>
> completing the request based on the version of the item and the version of the object.

The Examiner relies on the following as evidence of unpatentability:

Balabine	US 5,937,406	Aug. 10, 1999

Sue Plumley, *Ten Minute Guide to Windows NT Workstation 4.0* (Que, Aug. 1996) ("Windows NT").

David Pogue, *Mac OS X: The Missing Manual*, Ch. 2, at 63 (Pogue Press/O'Reilly & Assoc., 2d ed. 2002) ("Mac OS X").

Screenshots from Microsoft Windows XP Professional, Version 2002, Service Pack 1 ("About Windows: Confirm File Replace").[2]

[2] This reference's qualification as prior art is undisputed.

THE REJECTIONS

1. The Examiner rejected claims 1-4, 7, 9-16, and 18-20 under 35 U.S.C. § 103(a) as unpatentable over Windows NT and Balabine. Ans. 5-13.[3,4]
2. The Examiner rejected claims 8 and 17 under 35 U.S.C. § 103(a) as unpatentable over Windows NT, Balabine, and Mac OS X. Ans. 13-15.

CLAIM GROUPING

Appellants argue claims 1-4, 7, 9-16, and 18-20 together as a group, and also separately argue claims 18 and 19. *See* App. Br. 14-17; Reply Br. 4-6. Accordingly, we group claims 1-4, 7, 9-16, and 20 together, and select claim 1 as representative. We also treat claims 18 and 19 separately, and select claim 18 as representative of that group. *See* 37 C.F.R. § 41.37(c)(1)(vii).

THE OBVIOUSNESS REJECTION OVER WINDOWS NT AND BALABINE

Regarding representative claim 1, the Examiner finds that Windows NT discloses a method for validating requests with every recited feature

[3] Although the Examiner's rejection includes claims 5 and 6 (Ans. 5, 7), these claims have been cancelled. App. Br. 7 (noting that claims 5 and 6 have been cancelled). *Accord* Ans. 2 (confirming this status as correct). We therefore deem the Examiner's error harmless and present the correct claim listing here for clarity.

[4] Throughout this opinion, we refer to (1) the Appeal Brief filed October 23, 2007; (2) the Examiner's Answer mailed December 20, 2007; and (3) the Reply Brief filed February 19, 2008.

except that the validation is in connection with an interactive content database, but cites Balabine for that feature in concluding the claim would have been obvious. Ans. 5-6. In reaching this conclusion, the Examiner interprets the claimed "item" and "object" in two different ways.

First, the Examiner equates the "item" and "object" to two different computer files involved in a file replacement operation in Windows NT (i.e., older and newer versions of the same file). Ans. 15. The Examiner reasons that since the respective "versions" (i.e., the file creation dates) of the "item" and "object" would be identified and displayed to the user, the user would therefore complete the request to replace one file with the other based on this "version" information. Ans. 15-18.

Second, the Examiner equates the recited "item" to a folder, and the "object" to a file. Based on this alternative interpretation, the Examiner likewise finds that Windows NT's file or folder replacement functionality teaches completing a replacement request based on the identified versions of the item and object, respectively. *Id.*

Appellants argue that the Examiner's interpretation is flawed since an "item" cannot mean the same thing as an "object associated with the item." As such, Appellants contend that the cited prior art does not (1) identify a version of the item based on a first time; (2) determine whether the request affects an object associated with the item; (3) identify a version of the object based on a second time; and (4) complete the request based on the respective versions of the item and object as recited in claim 1. App. Br. 14-17; Reply Br. 4-6.

Appellants also contend that the cited prior art likewise fails to selectively delete an item and at least one associated object based on

whether a first age of the item is greater than or equal to a second age of the associated object as recited in claim 18. App. Br. 14; Reply Br. 6. Lastly, Appellants contend that the Examiner failed to adequately show why skilled artisans would have combined Windows NT with Balabine to arrive at the claimed invention. App. Br. 16-17; Reply Br. 7-9. The issues before us, then, are as follows:

ISSUES

1. Under § 103, has the Examiner erred by finding that Windows NT and Balabine collectively would have taught or suggested:

(1) (a) identifying a version of the item based on a first time; (b) determining whether the request affects an object associated with the item; (c) identifying a version of the object based on a second time; and (d) completing the request based on the respective versions of the item and object as recited in claim 1?

(2) selectively deleting an item and at least one associated object based on whether a first age of the item is greater than or equal to a second age of the associated object as recited in claim 18?

2. Is the Examiner's reason to combine the teachings of Windows NT and Balabine supported by articulated reasoning with some rational underpinning to justify the Examiner's obviousness conclusion?

FINDINGS OF FACT (FF)

1. According to Appellants' Specification, "[o]bjects may be any data entity for an item that is in digital form. For example, an object may be an audio file, an application, an image, text, or a video file." Spec. ¶ 022.

2. Appellants' Specification describes retrieving an object, such as a document or image file, that is referenced by an item. Spec. ¶ 020.

3. Appellants' Specification notes that objects can be associated with a particular item, such as a document. Spec. ¶ 028.

4. Windows NT describes using drag-and-drop to move or copy files and folders to new locations. With this feature, the user is warned when the user attempts to copy a file or folder to a location in which the file or folder with the same name exists. In this circumstance, a message displays the selected file's and the original file's size and creation or last modification date. The user can then click "Yes" to replace the file, or "No" to stop the process. Windows NT, at 1-2 ("Using Drag-and-Drop" section).

5. The "About Windows: Confirm File Replace" reference shows a Microsoft Windows screenshot of the contents of a folder "C:\Documents and Settings\bstace\Desktop\Test folder" which includes a file "sample.txt." "About Windows: Confirm File Replace," Fig. 3.[5]

6. The "Test folder" screen in the "About Windows: Confirm File Replace" reference includes a dialog box entitled "Confirm File Replace" that (1) indicates that the folder already contains a file named "sample.txt," and (2) asks whether the user wants to replace the existing 26-byte file that was modified August 18, 2006 at 1:45 PM with a 21-byte file that was

[5] Although the Examiner's obviousness rejections are based on Windows NT, the Examiner nonetheless cites the screenshots from Microsoft Windows ("About Windows: Confirm File Replace") to illustrate the functionality of commensurate file replacement features in Windows NT. *See, e.g.,* Ans. 16, 19-20. The Examiner's reliance on these screenshots for illustrative purposes is undisputed.

modified August 18, 2006 at 1:46 PM. The dialog box includes "Yes" and "No" buttons. *Id.*

7. Balabine's file system interface (IXFS 300) enables a database-unaware application 302 to access information in database 304 in a manner that is entirely transparent to the application by presenting contents of the database to the application as "file system objects" (e.g., directories, sub-directories, files, or links). Balabine, col. 5, ll. 5-32; col. 6, ll. 23-65; Figs. 3, 5C.

ANALYSIS

Claims 1-4, 7, 9-16, and 20

We begin by construing the key disputed limitations of claim 1 which call for, in pertinent part, an "item" and an associated "object." The Examiner construes these terms using two alternative interpretations. First, the Examiner equates the "item" and "object" to two different computer files involved in a file replacement operation in Windows NT (i.e., older and newer versions of the same file) (i.e., "Interpretation A" in the Examiner's parlance). Ans. 15. Alternatively, the Examiner equates the recited "item" to a folder and the "object" to a file (i.e., the "object" is a subset of the "item") ("Interpretation B"). *Id.*

Notably, Appellants dispute only the first interpretation (i.e., the "item" and "object" are files), for Appellants acknowledge that an "object" can be a subset of an "item," such as a files within folders (App. Br. 15)—an acknowledgement consistent with the Examiner's second interpretation. *Accord* Ans. 15 (referring to Appellants' acknowledgement on Page 15 of the Appeal Brief in connection with "Interpretation B").

Nevertheless, we find no error with either interpretation of "item" and "object" given the scope and breadth of the terms. Appellants' Specification defines "objects" quite broadly, indicating that they can be "*any* data entity for an item that is in digital form" such as files, applications, or text. FF 1 (emphasis added). Although Appellants' Specification does not squarely define "items," it does note that objects can be associated with items, such as *documents*. FF 3. We emphasize the word "documents" in connection with "items" here, for the same term is also used in connection with "objects" in the Specification. FF 2 (noting that an object can be a document).

These similarities are telling, for even assuming, without deciding, that an "object" and an associated "item" cannot be the same *unique instance* of a data entity, nothing in the claim nor the Specification precludes "objects" and "items" from referring to the same *type* of data entity, or even duplicate versions of the same data entity. That is, the limitations are fully met so long as an "object" (i.e., any data entity for an item that is in digital form (FF 1)) is somehow associated with an "item" which can be a similar—or even identical—version of that digital entity.

With this construction, we see no reason why an "item" and an associated "object" cannot refer to two different—yet associated—files, such as those involved in Windows NT's file replacement procedure (FF 4) as the Examiner indicates (Ans. 15). Although the selected and original files have the same name, they are nonetheless distinct—yet associated—entities since they are different versions of the same file—a distinction that is the very reason for warning the user that the original version will be replaced upon copying one file to the other's location. *See* FF 4-6.

Since the user is presented with the respective files' identities and creation or modification dates (which identify the versions of the files), the user's confirming file replacement by selecting "Yes" would complete the file replacement request based on the respective versions of the item and object. *See id.* That is, nothing in the claim precludes a determination that is based, at least in part, on a user's decision and affirmative action to replace an older version of a file with a newer version.

We reach a similar conclusion regarding the Examiner's alternative interpretation since, as the Examiner indicates (Ans. 17-18), skilled artisans would recognize that a folder ("item") would have a "version" based on the creation or modification dates of the files within that folder. As such, replacing a file (object) within that folder with a newer version would likewise affect the item (folder) not only in terms of its content, but also its "version" in light of this update. *See* FF 4-6. *Accord* Ans. 18. Accordingly, under the Examiner's alternative interpretation, the request would be completed based not only on the version of files (objects) within the folder (item), but also the version of the folder itself.

Lastly, we see no error in the Examiner's combining Balabine with Windows NT to arrive at the claimed invention. The Examiner cited Balabine merely to show that it would have been obvious to utilize Windows NT's request validation scheme in connection with interactive content databases. Ans. 6. We see no error in this reasoning, particularly since Balabine's file system interface presents database content to "database-unaware" applications as file system objects (e.g., directories, sub-directories, files, etc.) (FF 7)—the very objects that are represented in Windows for file management purposes. *See* FF 4-6. In short, using

Windows' request validation scheme for files and folders representing database content as suggested in Balabine is tantamount to the predictable use of prior art elements according to their established functions. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 417 (2007). We therefore find the Examiner's reason to combine the teachings of the cited references supported by articulated reasoning with some rational underpinning to justify the Examiner's obviousness conclusion.

We are therefore not persuaded that the Examiner erred in rejecting representative claim 1, and claims 2-4, 7, 9-16, and 20 which fall therewith.

Claims 18 and 19

We likewise sustain the Examiner's rejection of representative claim 18 which calls for, in pertinent part, processing requests that delete an item including an associated object, where an item *and* at least one associated object are selectively deleted based on whether a first age of the item is greater than or equal to a second age of the associated object essentially for the reasons indicated by the Examiner (Ans. 19-20).

Notably, unlike claim 1, claim 18 requires selectively deleting both the item *and* at least one associated object based on their relative ages. We see no error in the Examiner reliance on Windows NT and the associated "About Windows: Confirm File Replace" reference as teaching or suggesting these limitations, and we adopt the Examiner's findings and reasoning in this regard (Ans. 19-20) as our own. In particular, the Examiner's point that the respective versions of the "sample.txt" file on the desktop and in the "Test" folder would be deleted upon executing a file replacement via a drag-and-drop operation (*id.*) is well taken. *See* FF 4-6.

Although the user is presented with a message to confirm this replacement by clicking the "Yes" button as Appellants argue (Reply Br. 6), nothing in the claim precludes the user's involvement in selectively deleting the "item" and associated "object" based on the relative ages of those entities displayed as part of the confirmation message. *See* FF 4-6.

We are therefore not persuaded that the Examiner erred in rejecting representative claim 18, and claim 19 which falls therewith.

THE REJECTION OVER WINDOWS NT, BALABINE, AND MAC OS X

Regarding representative claim 8, the Examiner finds that Windows NT and Balabine disclose all recited features except for (1) determining if the version of the item is older than the version of the object, and (2) completing the request under that condition. The Examiner, however, cites Mac OS X for this feature in concluding the claim would have been obvious. Ans. 13-14, 20-22. Appellants argue that not only is there no reason to combine the references as proposed since Windows and Mac OS X are incompatible with each other, but Mac OS X fails to cure the deficiencies of Windows NT even if they were combinable. App. Br. 18. The issues before us, then, are as follows:

ISSUES

1. Under § 103, has the Examiner erred in rejecting claim 8 by finding that Windows NT, Balabine, and Mac OS X collectively would have taught or suggested (1) determining if the version of the item is older than the version of the object, and (2) completing the request under that condition?

2. Is the Examiner's reason to combine the teachings of these references supported by articulated reasoning with some rational underpinning to justify the Examiner's obviousness conclusion?

ADDITIONAL FINDINGS OF FACT

8. Mac OS X notes that if files are copied into a disk or folder that already contains items with the same names, the user is asked individually about each one. Moreover, Mac OS X 10.2 tells the user if the version being replaced is older or newer than the one that is moved. Mac OS X, at 63 (Tip under "Copying by Dragging" section).

ANALYSIS

We will sustain the Examiner's rejection of claim 8 essentially for the reasons indicated by the Examiner (Ans. 21-22). As the Examiner indicates (Ans. 21), the Examiner cited Mac OS X merely to show that it would have been obvious for the *computer* to determine the relative ages of the object and item in a replacement operation in lieu of the *user* as in Windows NT. We see no error in this approach since Mac OS X informs the user the relative ages of the files in a replacement operation. FF 8. Providing such an automatic functionality in the Windows NT would have been tantamount to the predictable use of prior art elements according to their established functions. *KSR*, 550 U.S. at 417.

Although Mac OS X is a different operating system than Windows NT, we nonetheless see no reason why skilled artisans could not have adapted the concepts from one operating system (e.g., Mac OS X) to another operating system (Windows) to obtain the relative advantages of each,

particularly since both systems involve file management operations, including file replacement. *Compare* FF 4-6 *with* FF 8. Such an enhancement has not been shown on this record to be beyond the level of ordinarily skilled artisans, and would therefore have been obvious. *See KSR*, 550 U.S. at 417 (noting that if a technique has been used to improve one device, and an ordinarily skilled artisan would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill).

We are therefore not persuaded that the Examiner erred in rejecting representative claim 8, and claim 17 which falls therewith.

CONCLUSION

The Examiner did not err in rejecting claims 1-4 and 7-20 under § 103.

DECISION

The Examiner's decision rejecting claims 1-4 and 7-20 is affirmed.

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED

msc

SUGHRUE MION PLLC
USPTO CUSTOMER NO WITH IBM/SVL
2100 PENNSYLVANIA AVENUE, N.W.
WASHINGTON DC 20037